

Unidad I: Introducción a C#

Lic. Ronaldo Armando Canizales Turcios

Departamento de Electrónica e Informática, UCA

Ciclo virtual 01-2021

Agenda

- 1 Historia
- 2 Nuestro primer programa
- 3 Generalidades
- 4 Sentencias de control de flujo
- 5 Generación de ejecutables (.exe)
- 6 Anuncios

1. Historia

Breve historia de C#

C# es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. Su sintaxis básica deriva de C/C++ aunque incluye mejoras derivadas de otros lenguajes. C# es un lenguaje de programación diseñado para generar programas sobre la plataforma .NET. Existe un compilador que provee el marco Mono-DotGNU, el cual genera programas para distintas plataformas como Windows Microsoft, Android, iOS, Windows Phone, Mac OS y GNU/Linux.

En abril de 1999, **Anders Hejlsberg** formó un equipo con la misión de desarrollar un nuevo lenguaje orientado a objetos, C#, la primera versión fue publicada en enero del 2000.

2. Nuestro primer programa

Hola Mundo en C#

```
using System;  
  
class HolaMundo {  
    static void Main() {  
        Console.WriteLine("Hola mundo!");  
    }  
}
```



3. Generalidades

Similitudes con C++

- Comentarios: de línea (`//`) y de bloque (`/* */`).
- Sintaxis para declaración de variables (tipo nombre = valorInicial;).
- Nombres de tipo de variable (int, char, double, float, long).
- Mostrar en consola con y sin salto de línea (`Write` y `WriteLine`).
- Operadores lógicos `AND(&&)`, `OR(||)` y `NOT(!)`.
- Operadores de comparación (`==`, `<`, `<=`, `>`, `>=`, `!=`).

Diferencias con C++

- Nombres de algunos tipos de variables (string, bool).
- Solicitar datos en consola (`Console.ReadLine()`) es necesario realizar conversiones mediante la clase estática `Convert`.
- Manipulación y operaciones de cadenas de caracteres.

3. Generalidades

Primera gran diferencia con C++

Un paso muy importante en todo programa es la interacción con el usuario, este paso se hace de manera un tanto distinta en C# y C++. En el caso de C# **siempre se obtienen cadenas de texto** y luego éstas se convierten al tipo de dato que nos interesa.

```
String inputCadena = "";  
inputCadena = Console.ReadLine();  
  
int inputEntero = 0;  
inputEntero = Convert.ToInt32(Console.ReadLine());  
  
double inputDouble = 0.0;  
inputDouble = Convert.ToDouble(Console.ReadLine());
```



3. Generalidades

Manipulación de cadenas de caracteres (string)

- Longitud (cantidad de caracteres): **variable.Length**
- Convertir a mayúsculas: **variable.ToUpper()**
- Convertir a minúsculas: **variable.ToLower()**
- Acceso a alguna letra en específico: **variable[índice]**
- Obtener subcadenas: **variable.Substring(índice)**
- Índice de un caracter en específico: **variable.IndexOf(caracter)**

- Comparar: **variable1.CompareTo(variable2)**
- Remover espacios en blanco: **variable.Trim()**
- Separar en subcadenas basado en un caracter delimitador: **variable.Split(caracter)**



3. Generalidades

Ejemplo: Datos carné UCA

Solicite los siguientes datos al usuario y luego muestre un resumen.

- Nombre: cadena de texto.
- Apellido: cadena de texto.
- Edad: número entero.
- Carrera: cadena de texto.



4. Sentencias de control de flujo

La sintaxis es prácticamente la misma en C# y C++

- **if simple**: hacer algo si se cumple una condición
- **if-else**: hacer algo si es cierto u otra cosa si es falso.
- **if-else-if**: más de una prueba, se permite anidamiento.
- **switch**: para variables enteras y de carácter.
- Operador ternario: (**condición ? siVerdadero : siFalso**).
- **while** y **do-while**: repetir mientras se cumple una condición.
- **for**: repetir una cantidad específica de veces.
- Incrementos y decrementos: $i++$, $i--$, $i+=2$, $i-=2$, $i*=2$, etc.

Ejemplo: Precio entradas Rusia 2017

Solicite al usuario la cantidad de entradas a comprar y el correlativo del partido al que desea asistir, realice las validaciones necesarias. Finalmente calcule el precio en Rublos (categoría 4).

¿Y si hubiera un impuesto del 13 %?



PARTIDO	PRECIOS EN USD			PRECIOS EN RUB
	CAT 1	CAT 2	CAT 3	CAT 4*
PARTIDO INAUGURAL (N° 1)	170	120	85	2 240
FASE DE GRUPOS (N° 2 A 12)	135	85	70	960
SEMIFINALES (N° 13 A 14)	160	125	85	2 240
PARTIDO POR EL 3.º Y 4.º PUESTO (N° 15)	135	85	70	960
FINAL (N° 16)	245	160	115	2 560

*Solo para residentes en Rusia



5. Generación de ejecutables (.exe)

- 1 Disco local (C:).
- 2 Usuarios.
- 3 **Tu usuario en específico.**
- 4 RiderProjects.
- 5 **El proyecto en específico.**
- 6 Carpeta de nombre del proyecto.
- 7 Carpeta “bin”.
- 8 Carpeta “Debug”.
- 9 Listo, se puede ejecutar con cmd (consola de Windows).



6. Anuncios

Generalidades

- Estar siempre pendientes del Moodle.
- Al mandar un correo, se debe anteponer [0121POO] en el asunto.
- Contacto: rcanizales@uca.edu.sv & 00019618@uca.edu.sv

Semana 1

- Unirse al canal en Discord (opcional, pero recomendado).
- Inscribir clanes (parejas de trabajo) en Moodle.
- Instalar IDE en computadora personal.
- Realizar actividades optativas que les llamen la atención.
- Asistir a la práctica de laboratorio (Walter hará la convocatoria).

7. Referencias

- J. Albahari y B. Albahari, C# 7.0 in a Nutshell. Editorial O'REILLY, 2018.
- J. Albahari y B. Albahari, C# 7.0 Pocket Reference. Editorial O'REILLY, 2018.
- B. Wagner, More effective C#. Editorial Addison-Wesley Professional, 2017.
- J. Bloch, Effective Java. Editorial Addison-Wesley, 2008.
- C. Horstmann y G. Cornell, Core Java. Editorial Prentice Hall, 2007.
- C. Horstmann, Computing concepts with Java Essentials. Editorial WILEY, 2003.
- B. Meyer, Construcción de software orientado a objetos. Editorial Prentice Hall, 1997.

Unidad I: Introducción a C#

Lic. Ronaldo Armando Canizales Turcios

Departamento de Electrónica e Informática, UCA

Ciclo virtual 01-2021